

Perancangan dan Implementasi Aplikasi Android Streaming (Studi Kasus FTI Universitas Kristen Satya Wacana Salatiga)

Radius Tanone^{#1}, Sushendra Ipol^{*2}

[#]Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana
Jl. Diponegoro 52-60, Salatiga

^{#1}radius.tanone@staff.uksw.edu

^{*} ²672009275@staff.uksw.edu

Abstract— FTI UKSW memiliki kegiatan-kegiatan baik yang bersifat kegiatan kemahasiswaan maupun kegiatan umum yang ada di fakultas. Peserta yang mengikuti kegiatan tersebut terdiri dari mahasiswa, dosen dan khalayak umum. Kesibukan atau tidak memiliki kesempatan untuk mengikuti acara tersebut secara langsung dapat menjadi kendala karena tidak mendapatkan kesempatan mengikuti acara tersebut. Berdasarkan masalah yang ada dilakukan penelitian dengan menggunakan tahapan identifikasi masalah, tinjauan pustaka, perancangan sistem dan penulisan laporan untuk membangun sebuah aplikasi android *streaming* untuk pengambilan gambar. Selain itu juga dibangun sebuah *web* untuk menghasilkan output dari *streaming* yang ada kepada *user*. Hasil dari penelitian ini adalah menghasilkan aplikasi android *streaming* yang dapat digunakan oleh pihak fakultas sehingga masalah kesibukan atau tidak dapat hadir langsung ditempat dapat terjawab. Aplikasi ini dapat membantu pihak yang berkepentingan agar tetap mendapat kesempatan mengikuti acara secara realtime dengan *streaming* menggunakan mobile android pada sisi pengambilan gambar dan *web* untuk melihat hasil *streaming*.

Keywords— *Android Streaming, RTMP, RTSP.*

I. PENDAHULUAN

Seminar dan acara-acara fakultas merupakan kegiatan yang sering dilakukan oleh Fakultas Teknologi Informasi Universitas Kristen Satya Wacana Salatiga (FTI-UKSW). Kegiatan yang dilakukan antara lain seminar mahasiswa dan dosen, acara yudisium fakultas dan lain sebagainya. Begitu banyaknya kegiatan serta tidak diimbangi dengan waktu dan kesempatan untuk menghadiri dapat membuat kegiatan-kegiatan tersebut terlewat. Kesibukan mahasiswa maupun dosen merupakan salah satu kendala yang sering terjadi sehingga tidak bisa mengikuti kegiatan yang sedang berlangsung.

Seiring dengan perkembangan teknologi dan bertambahnya produk *smartphone* yang sangat pesat juga para penyedia *provider* maka dapat menyediakan layanan

internet untuk memudahkan setiap penggunanya mendapatkan informasi yang diinginkan dimana saja dan kapan saja. Perkembangan teknologi *streaming* saat ini juga merupakan salah satu yang dapat digunakan untuk berbagi informasi kepada *user* yang menggunakannya. Untuk menjawab masalah di atas maka penerapan teknologi *streaming* di *smartphone* dapat menjadi solusi bagi mahasiswa dan dosen serta pihak yang berkepentingan untuk mendapatkan informasi walaupun tidak berada di tempat kegiatan.

Penerapan android *streaming* yang dimaksud yaitu menggunakan protokol Real Time Messaging Protocol (RTMP) dan Real Time Streaming Protocol (RTSP) pada proses *streaming* yang berlangsung. *Server* RTSP dibangun untuk berinteraksi dengan perangkat android *stream video* dan aplikasi [1]. Adanya RTMP dan RTSP membuat aplikasi dapat dibangun di *smartphone* android sehingga dapat membagi informasi ke *user*.

Rumusan masalah yang didapatkan dari latar belakang masalah adalah bagaimana membagi informasi mengenai kegiatan-kegiatan yang dilakukan di FTI UKSW dan bagaimana mengikuti kegiatan-kegiatan secara real time menggunakan android *streaming*. Adapun tujuan penelitian ini adalah membuat android *streaming* dengan memanfaatkan Real Time Messaging Protocol (RTMP) dan Real Time Streaming Protocol (RTSP) untuk berbagi informasi mengenai kegiatan yang dilakukan dan memudahkan dosen dan mahasiswa bisa mengambil bagian dari kegiatan-kegiatan yang dilakukan walaupun tidak ikut serta dalam dalam kegiatan sedang berlangsung. Sehingga manfaat yang didapatkan dari penelitian ini adalah menghasilkan aplikasi android *streaming* yang dapat mempermudah mahasiswa dan dosen untuk bisa mendapatkan informasi-informasi dari kegiatan-kegiatan yang sedang dan akan dilakukan oleh FTI UKSW.

II. TINJAUAN PUSTAKA

Penelitian sebelumnya dilakukan dengan menyajikan pembuatan aplikasi *video streaming* di ponsel melalui koneksi WLAN menggunakan teknologi JSR 135 atau lebih dikenal dengan Mobile Media API (Mmap). Mmap digunakan untuk mengontrol proses mendukung *video streaming* dan fitur Its. Aplikasi ini akan menggunakan dua protokol yaitu RTSP dan HTTP. Kelebihan dari penelitian ini adalah waktu yang diperlukan untuk memutar *file multimedia* tidak terpengaruh oleh ukuran *video file streaming* [2]. Adapun penelitian lain yang membahas pembangunan aplikasi *video streaming* berbasis android di stv Bandung dengan teknologi broadcast hingga terbentuk gabungan teknologi *streaming* dengan teknologi mobile berbasis android. Dalam pengembangan aplikasi *video streaming* juga dilakukan tahapan untuk mengetahui kelebihan dan kekurangannya dengan pengujian menggunakan parameter *Quality of Service* (QoS) seperti berapa lama *delay* dan kualitas yang dihasilkan dari aplikasi *video streaming* ini [3].

Sehingga dengan ini maka dikembangkan aplikasi android *streaming* pada perangkat android yang akan menjadi pencipta *video* melalui kamera dan bertanggung jawab untuk transmisi *video* melalui internet ke *server*. Hal ini juga mempertahankan komunikasi dengan *server* melalui protokol kustom untuk memastikan identifikasi dan *video streaming* [1]. Saat ini platform android sudah dapat digunakan untuk *video streaming*.

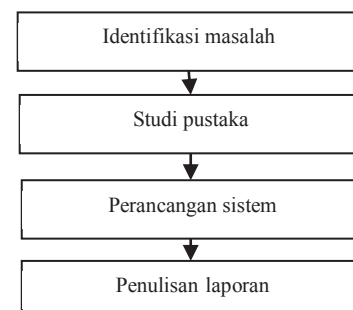
Real Time Messaging Protocol (RTMP) adalah aplikasi lapisan *proprietary* protokol yang dikembangkan oleh *Adobe Systems* [4]. *Real Time Messaging Protocol* (RTMP) juga menyediakan dua arah yang pesan layanannya *multiplex* seperti TCP [RFC0793] yang dimaksudkan untuk membawa aliran paralel *video*, audio, dan pesan data, dengan informasi waktu yang terkait [5]. Protokol RTMP memiliki beberapa variasi; protokol "plain" yang bekerja di atas dan menggunakan TCP nomor port 1935 secara default. RTMPS yang RTMP melalui koneksi TLS / SSL, RTMPE yang RTMP dienkripsi menggunakan mekanisme keamanan *Adobe* sendiri [5].

Real Time Streaming Protocol (RTSP) adalah sebuah protokol level aplikasi untuk kontrol atas pengiriman data dengan sifat *real-time*. RTSP menyediakan kerangka *extensible* untuk mengaktifkan kendali pada pengiriman data *real-time*, seperti *audio* dan *video*, dengan menggunakan *Transmission Control Protocol* (TCP) atau *User Data Protocol* (UDP). Proses kerja *Real Time Streaming Protocol* adalah protokol level aplikasi yang bertujuan untuk menyediakan sebuah protokol yang kuat untuk multimedia *streaming* satu ke banyak aplikasi secara unicast dan multicast serta untuk mendukung interoperabilitas antara client dan *server* dari vendor yang berbeda. RTSP dianggap lebih dari kerangka daripada protokol. RTSP didesain untuk bekerja di atas RTP untuk mengontrol dan menyampaikan konten secara *real-time*.

Protokol RTSP menempatkan semua metode secara bersama-sama. Mulai dari mengirim permintaan kontrol, membangun sebuah line, permintaan URL, dan nomor versi protokol. Kemudian klien termasuk *general header*, *header* permintaan dan *header* entitas membentuk protokol http. Ini yang kemudian dikirim ke *server* yang akan mengeksekusi permintaan jika memungkinkan. Kemudian *server* merespon status-line, *general header* dan *header* entitas. Status line berisi versi protokol, kode status numerik dan deskripsi tekstual. Media *stream* yang tersisa tidak ditentukan oleh RTSP. Hal ini bisa merupakan *streaming* RTP atau bentuk lain dari transmisi media. RTSP hanya menetapkan kontrol dan sampai dengan perangkat lunak klien dan *server* untuk menjaga pemetaan antara saluran kontrol dan media *stream*. Sebuah konsep kunci dalam RTSP adalah gagasan tentang *session*. RTSP bekerja dengan terlebih dahulu meminta presentasi yang akan dimulai oleh *server*, menerima *session identifier* yang kemudian digunakan dalam semua kontrol berikutnya. *Session* pengidentifikasi merupakan *state* bersama antara *client* dan *server*. Permintaan kontrol dan tanggapan kemudian dapat dikirim melalui TCP [6].

III. METODE PENELITIAN

Metode penelitian yang digunakan adalah deskriptif kualitatif. Karena bersifat deskriptif maka penekanannya pada proses, makna, dan pemahaman melalui kata-kata atau gambar dan bersifat induktif dengan membangun abstraksi, konsep-konsep, hipotesis, dan teori secara rinci. Karakteristik dari penelitian kualitatif adalah fokus terhadap proses dibandingkan hasil yang terdiri dari instrument utama dalam pengumpulan dan analisa data dengan melibatkan kerja lapangan [7]. Penelitian dilakukan melalui identifikasi masalah, studi pustaka, perancangan sistem dan penulisan laporan. Subjek penelitian adalah dosen dan mahasiswa yang berhalangan ikut kegiatan dikarenakan terbatasnya ruangan, tempat dan kesibukan yang terjadi pada dosen dan mahasiswa serta terlambatnya mahasiswa untuk mendaftar. Perangkat aplikasi yang berhubungan dengan *video streaming* yang memanfaatkan RTMP dan RSTP. Adapun tahapan-tahapan penelitian yang dilakukan dapat dilihat pada Gambar 1, yaitu:



Gambar 1 Tahapan Penelitian

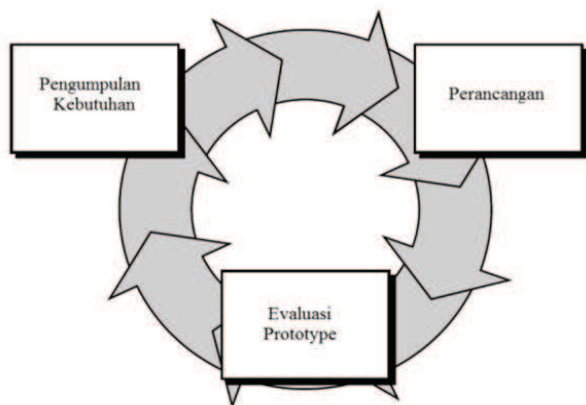
Tahap Pertama adalah identifikasi masalah, dalam tahap penelitian ini dilakukan identifikasi masalah-masalah yang terjadi pada FTI UKSW berupa wawancara kepada mahasiswa dan dosen yang memiliki masalah kesibukan dan terbatasnya kapasitas tempat pelaksanaan kegiatan-kegiatan yang dilakukan dan lain sebagainya.

Tahap kedua adalah studi pustaka, membaca referensi-referensi, penjelasan penelitian-penelitian sebelumnya dan pelajari teori-teori yang berkaitan dengan penelitian. Hal ini dipakai untuk menjawab masalah dengan membangun aplikasi android *streaming*.

Tahap ketiga adalah perancangan sistem. Pada tahap ini dilakukan proses perancangan sistem (aplikasi android *streaming*) menggunakan *model prototype* untuk mengetahui setiap proses beserta semua aktifitas yang akan dibangun pada sistem.

Pada tahap keempat yang dilakukan adalah penulisan laporan. Pada tahap ini menjelaskan latar belakang masalah, rumusan masalah, batasan masalah, tinjauan pustaka, metode penelitian, pembahasan, kesimpulan dan saran serta daftar pustaka.

Tahapan *prototype* model ditunjukkan pada Gambar 2. *Prototype* model merupakan perancangan sistem yang digunakan untuk pengumpulan persyaratan aplikasi, pembangunan aplikasi dan evaluasi aplikasi yang sudah dibangun apakah sudah sesuai dengan kebutuhan *user*.



Gambar 2 Prototyping Model [7]

Tahapan *prototype* model diawali dengan pengumpulan kebutuhan yaitu mencari kebutuhan keseluruhan perangkat aplikasi yang berhubungan dengan *video streaming*. Dalam tahap pengumpulan kebutuhan sistem merupakan proses identifikasi dan meneliti permasalahan-permasalahan yang ada, sehingga sistem yang dibangun nanti sesuai dengan kriteria kebutuhan yang diperlukan dalam simulasi untuk mencapai tujuan dibuatnya *video streaming* dan android sebagai *mobile device*. Pada tahap pengumpulan kebutuhan ini, perangkat keras dan perangkat lunak komputer dalam pembuatan *video streaming* adalah handphone android dan PC/Laptop (*hardware*) sedang *software* yang digunakan adalah OS: API Maximum 19, API Minimum 8, JW Player dan *WowzaMedia Engine* sebagai *server*.

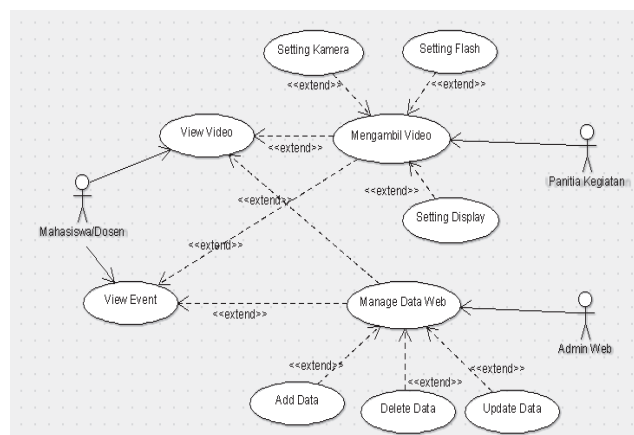
Tahapan *prototype* model yang kedua dimulai dengan perancangan UML untuk proses perancangan aplikasi mobile dan *web* untuk *user*. Proses berikutnya adalah instalasi *wowza* sebagai media *engine* dan juga menggunakan *lib streaming* untuk transfer data atau gambar yang diminta *user* serta pemasangan JW Player pada *web*.

Tahapan *prototype* model yang ketiga adalah Evaluasi. Evaluasi dilakukan dengan pengujian *alfa* dan pengujian *beta*. Evaluasi tahap pengujian alfa dilakukan setelah rancangan awal program selesai dibuat. Dalam tahap ini juga untuk mengecek apakah menu-menu dari tampilan program dapat dijalankan atau tidak sesuai dengan fungsinya masing-masing.

Evaluasi tahap *beta* disesuaikan dengan menambahkan fungsi-fungsi yang dapat mengembangkan program agar dapat memudahkan dosen dan mahasiswa untuk melihat kegiatan-kegiatan yang berlangsung walaupun tidak ikut serta dalam kegiatan tersebut.

Diagram use case merupakan *diagram* yang menjelaskan manfaat sistem jika dilihat dari sudut pandang orang atau sesuatu yang berada diluar sistem yang sedang dibangun (aktor). Jenis *diagram* ini dapat digunakan untuk menangkap *requirements* sistem dan untuk memahami bagaimana sistem seharusnya bekerja [8].

Berdasarkan sistem maka ditentukanlah *user* dalam hal ini mahasiswa dan dosen yang akan melihat *video streaming* dan *event* kegiatan dan *user* kameramen (panitia kegiatan) seperti yang terlihat pada Gambar 3.

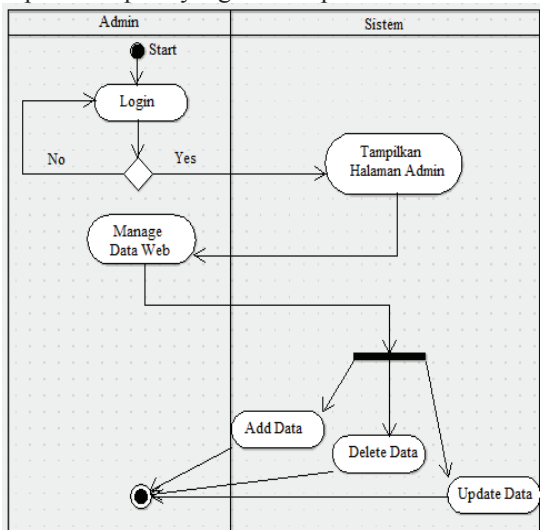


Gambar 3 User Case Diagram Android Streaming

Pada *usecase diagram* terdapat 1 aktor *admin* yaitu *admin web* dan 2 aktor yaitu *user* klien (mahasiswa dan dosen) dan *user* kameramen (panitia kegiatan yang menggunakan aplikasi di android) dalam perancangan aplikasi seperti yang terlihat pada Gambar 3. Aktor *admin* bertugas untuk *manage data web* dengan *add* data, *hapus* data, *update* data, *lihat video* dan *event*. Aktor *user* dapat *lihat video* dan *event* sedangkan *user* kameramen *mengambil video* agar *video* dapat dilihat oleh *user* melalui *webstream*. *User 2* juga dapat melakukan proses *setting display*, *setting flash* dan *setting kamera*.

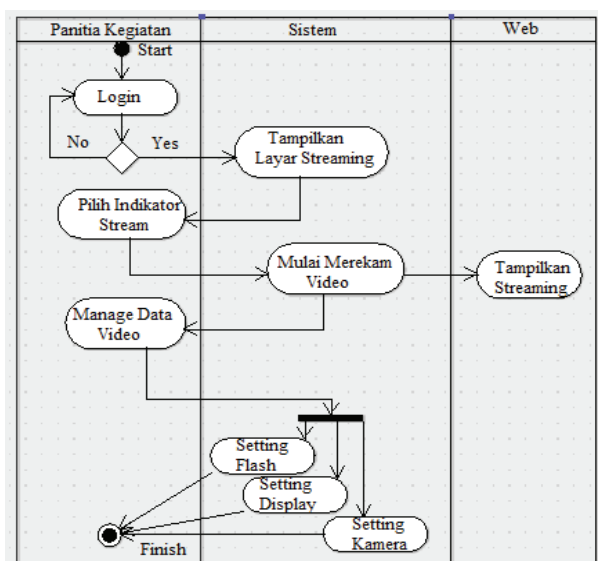
Diagram aktivitas adalah *diagram* yang memodelkan alur kerja (*workflow*) sebuah proses bisnis dan urutan aktivitas langkah per langkah dalam suatu proses [8].

Admin mempunyai beberapa hak yang dapat dilakukan pada aplikasi seperti yang terlihat pada Gambar 4.



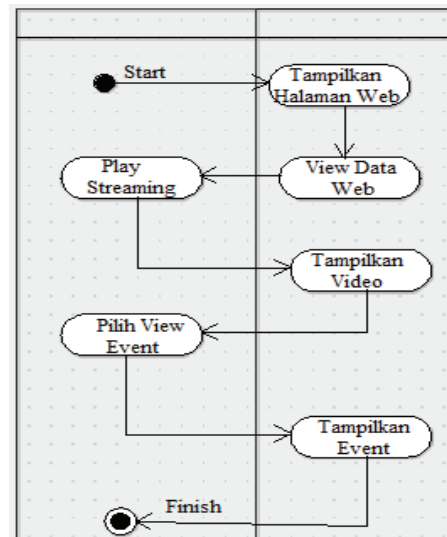
Gambar 4 Activity Diagram Admin

Gambar 4 merupakan *activity diagram* yang dimiliki oleh seorang *admin* (*admin web*). *Admin* dapat melakukan pengelolaan data *web* dengan menambahkan data (*add data*), menghapus data (*delete data*) dan memperbaharui data (*update data*).



Gambar 5 Activity Diagram User Kameramen

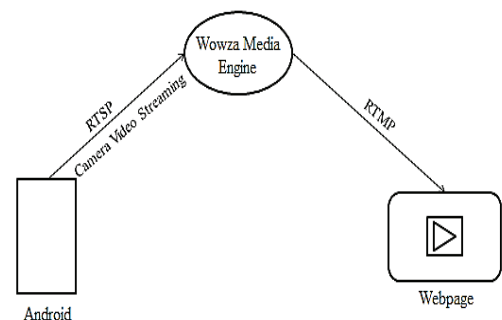
Gambar 5 merupakan *activity diagram* yang dimiliki oleh *user kameramen* (*panitia kegiatan*). *User kameramen* melakukan pengambilan *video* agar *video* yang direkam dapat ditampilkan di *web*. *User kameramen* juga dapat melakukan pengelolaan data *video* dengan mengatur *flash*, mengatur *display* dan mengatur *kamera*.



Gambar 6 Activity Diagram User Client

Gambar 6 adalah *activity diagram* *user* klien (*mahasiswa dan dosen*). *User* klien hanya dapat melakukan *view data web* yaitu hanya dapat melakukan *play streaming*, *view video* dan *view event*.

Arsitektur sistem protokol *video streaming* dengan menggunakan *RTMP* dan *RSTP* dapat dilihat pada Gambar 7 sebagai berikut;

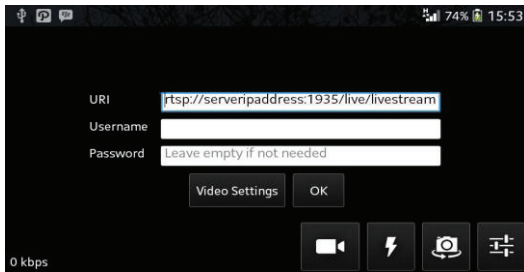


Gambar 7 Arsitektur Sistem [9]

Gambar 7 adalah arsitektur sistem android *streaming* yang menjelaskan bahwa saat android mulai menjalankan program, camera mulai mengambil *video*. *Video* dikirim ke wowza media engine sebagai mesin *streaming* yang dapat melakukan *streaming video* dan wowza media engine juga bertindak sebagai kerangka *streaming server* yang menerima *video* dari perangkat android menggunakan protokol *RTSP* dengan menggunakan port 1935. Setelah *stream* data diterima oleh media engine, data *stream* akan di-decode dan ditampilkan ke *webpage* menggunakan protokol *RTMP*.

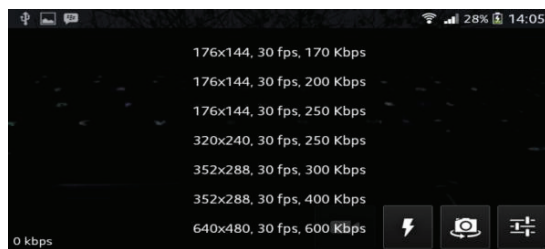
IV. PEMBAHASAN

Berikut adalah hasil dan pembahasan dari pengembangan aplikasi android *streaming*.



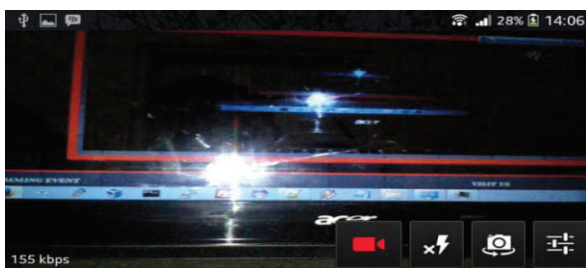
Gambar 8. Tampilan Awal Saat Aplikasi Android *Streaming* Dijalankan

Gambar 8 merupakan tampilan pertama kali ketika aplikasi dijalankan (setelah *user* melakukan *login*), aplikasi akan langsung meminta konfigurasi untuk URL, *user* diminta untuk memasukan *username* dan *password* agar dapat terhubung dengan *server*.



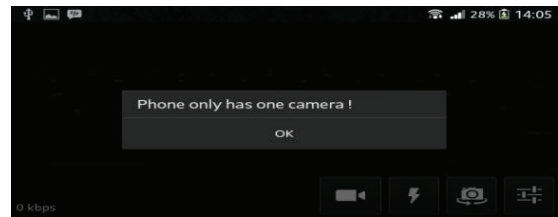
Gambar 9 Video Setting

Gambar 9 merupakan tampilan yang muncul jika *user* menekan *video setting*. Pada fungsi ini akan menampilkan resolusi yang dipilih oleh *user* dan *user* dapat memilih resolusi sesuai keinginan *user*.



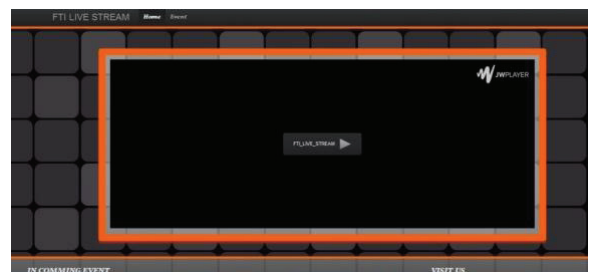
Gambar 10 Konfigurasi *Server* Benar

Gambar 10 menjelaskan jika konfigurasi terhadap *server* benar maka aplikasi dapat memulai *streaming video* dengan indikator logo *stream* berubah menjadi merah. Logo petir dengan tanda silang merupakan indikator *flash* sedang dijalankan jika *user* ingin mematikan *flash* dapat diakses dengan menekan tombol tersebut. Gambar 9 menjelaskan *device* kamera yang digunakan.



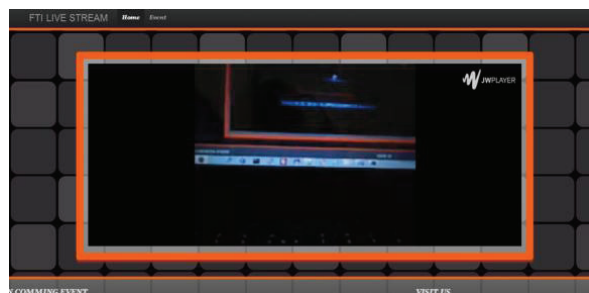
Gambar 11. *Camera Device* pada *Smartphone*

Gambar 11 menjelaskan pada tombol kamera, *user* dapat mengakses *device* kamera apakah memiliki *front* kamera atau tidak. Jika kamera *user* tidak memiliki *front* kamera maka akan muncul popup yang menerangkan bahwa *device* hanya memiliki 1 kamera saja. Tampilan awal *web* dapat terlihat pada Gambar 12.



Gambar 12 Tampilan Awal *Web*

Pada halaman home *web* ini *user* dapat melihat data *web* yaitu dengan melihat *video* dan event-event yang sedang berlangsung maupun yang akan berlangsung. Halaman home *web* ini juga merupakan tampilan awal *web*, dimana tampilan awal *web* yang menampilkan home live *streaming*, *event* dan *incoming event*. Gambar 13 menjelaskan proses *streaming* berlangsung.



Gambar 13 Proses *Streaming* Berlangsung

Pada Gambar 13 merupakan tampilan *web* ketika proses *streaming* berlangsung. Sementara proses *streaming* berlangsung *user* dapat melihat kegiatan yang dilakukan FTI. *Streaming* yang terjadi terdapat delay selama 2-4 detik. Pada halaman *web* ini, data event yang berlangsung di *add*, *edit* dan *delete* oleh *admin web*.

Pengujian dilakukan dengan menggunakan pengujian *alfa* dan *beta*. Pengujian *alfa* dapat dilihat pada Tabel 1.

TABEL 1 PENGUJIAN ALFA

No	Masukan Yang Dilakukan	Keluaran Yang Diharapkan	Hasil	Kesimpulan
1	URL salah server	-Input berhasil	-Gagal tersambung	Valid
2	URL benar sever	-Input berhasil	- Tersambung dengan server	Valid
3	Username dan password salah	-Input berhasil	-Gagal tersambung	Valid
4	Username dan password benar	-Input berhasil	- Tersambung dengan server	Valid
5	Memilih salah satu video setting	-Device Support	-Muncul popup pilhan	Valid
6	Memilih salah satu video setting	-Device tidak support	-Muncul pesan kesalahan	Valid
7	Menekan button flash	Input berhasil	-Flash pada device menyala	Valid
8	Menekan button flash	-Device tidak memilki flash	-Muncul pesan error	Valid
9	Menekan button kamera	-Input berhasil	-Berubah ke kamera depan	Valid
10	Menekan button kamera	-Device tidak memiliki kamera depan	-Muncul pesan error	Valid

Pengujian dilakukan terhadap beberapa dosen, staff dan mahasiswa FTI, dengan jumlah responden 40 orang. Pengujian dilakukan dua cara yaitu pengujian alfa dan pengujian beta. Pengujian alfa dilakukan oleh pembuat aplikasi untuk sirkulasi internal dan *error* atau ketidaklengkapan dalam aplikasi. Dari Tabel 1 disimpulkan bahwa semua variabel dalam aplikasi dapat berjalan dengan baik dan siap untuk diujikan kepada target pegujian aplikasi. Pengujian beta dilakukan dengan membagi kusioner kepada responden seputar aplikasi Android *Streaming* untuk mengetahui pendapat responden terhadap aplikasi.

Dalam tahap pengisian kusioner ada lima kategori pertanyaan dimana kategori pertama 95% responden menjawab aplikasi android *streaming* sangat membantu responden dalam mengikuti kegiatan secara *streaming*. Kategori kedua sebanyak 90% responden berpendapat aplikasi android *streaming* dapat memudahkan responden untuk mendapatkan informasi mengenai kegiatan-kegiatan yang dilakukan FTI UKSW. Pada kategori ketiga sebanyak 97,5% responden berpendapat bahwa *event* yang ada pada aplikasi android *streaming* dapat membantu untuk mengikuti kegiatan dan event dengan keterbatasan waktu yang responden miliki. Kategori keempat sebanyak 97,5% responden berpendapat bahwa aplikasi android *streaming* mudah dimengerti dan digunakan. Pada kategori kelima yaitu sebanyak 95% responden sangat setuju dengan interface aplikasi android *streaming* ini. Dari kelima kategori tersebut dua kategori menghasilkan presentase 97,5% sedangkan dua kategori menghasilkan presentase 95% dan satu kategori menghasilkan presentase 90%. Secara keseluruhan 95% responden mengatakan aplikasi android *streaming* dapat membantu para mahasiswa maupun dosen secara *streaming* untuk mengikuti kegiatan FTI UKSW. Sehingga dari data-data hasil kusioner menunjukan aplikasi android *streaming* menjawab kebutuhan *user*.

V. KESIMPULAN

Kesimpulan yang dapat diambil dari hasil penelitian yang telah dilakukan bahwa aplikasi android *streaming* dapat mengatasi masalah yang ada dengan memudahkan mahasiswa dan dosen untuk mengikuti kegiatan secara *realtime* pada saat proses kegiatan sedang berlangsung tanpa harus datang langsung ke tempat pelaksanaan. Aplikasi android *streaming* ini juga dapat membagi informasi mengenai kegiatan-kegiatan yang dilakukan FTI UKSW kepada mahasiswa dan dosen sehingga apabila ingin mengikutinya maka dapat menggunakan dengan aplikasi android *streaming* ini. Adapun saran yang disampaikan dari penelitian ini adalah selain menggunakan *platform* android bisa dikembangkan menggunakan aplikasi *multiplatform mobile*.

DAFTAR PUSTAKA

- [1] Bailey, Justin., 2011., *Live Video Streaming from Android-Enabled Devices to Web Browsers*, Graduate Theses and Dissertations., University of South Florida Scholar Commons.
- [2] Shiddiq, Ary Mazharuddin., Pratama, Henry., Ciptaningtyas, Henning Titi., 2010., *A Video Streaming Application Using Mobile Media Application Programming Interface*.
- [3] Arsam, Arfiandy, 2012, *Pembangunan Aplikasi Video Streaming Berbasis Android di STV Bandung*, Jurnal Ilmiah Komputer dan Informatika (KOMPUTA).
- [4] Seeling, Patrick., 2011., *Web Conferencing Traffic-An Analysis Using Dimdim As Example.*, *International Journal of Computer Networks & Communications (IJCNC)* Vol.2, No.6:2.
- [5] Wikipedia., 2014., *Real Time Messaging Protocol* “., http://en.wikipedia.org/wiki/Real_Time_Messaging_Protocol., 18 Januari 2015.

- [6] Satwika, Kadek, Susila., 2011., *Proses Video Streaming Dengan Protocol Real Time Streaming Protocol (Rtsp).*, Pengolahan Sinyal Multimedia Te(6610).
- [7] Pressman, Roger S., 2001., *Software Engineering A Practitioner's Approach.*, Newyork: Published by McGraw-Hill, an imprint of The McGraw-Hill Companies, Inc.
- [8] Sasongko, Agus., 2009., "Unified Modeling Language (UML)"uml,http://agoenxz21.blogspot.com/2009/10/unified-modeling-language-uml.html., diakses 17 Januari 2015
- [9] Tamada, Ravi., 2014., "Android Streaming Live Camera Video to Web Page",. http://www.androidhive.info/2014/06/android-streaming-live-camera-video-to-web-page/, diakses 18 Januari 2015.